基于标准置换检验的差异序列模式挖掘算法 *

吴 军，欧阳艾嘉，张 琳

(遵义师范学院 信息工程学院，贵州 遵义 563000)

摘 要：为了去除差异序列模式挖掘算法返回结果中的假阳性差异序列模式，提出了一个基于标准置换假设检验的算法 SP-DSP。该算法首先运用 GSP 算法挖掘频繁序列模式，然后基于 Growth rate 阈值生成差异序列模式候选集，并运用标准置换检验计算候选集合中每个模式的 p-value，最后运用多重假设检验度量过滤假阳性差异序列模式。实验结果证明 SP-DSP 算法能够去除掉一定数量的假阳性模式并尽可能地保留真差异序列模式，从而促进后续分类任务正确率的提升。

关键词：差异序列模式挖掘；模式评估；多重假设检验；标准置换检验

中图分类号：TP391 doi: 10.19734/j.issn.1001-3695.2020.04.0058

Mining discriminative sequential patterns based on standard permutation testing

Wu Jun, Ouyang Aijia, Zhang Lin

(*School of Information Engineering*, *Zunyi normal university*, *Zunyi Guizhou* 563000, *China*)

Abstract: To filter out the false positive patterns returned from the discriminative sequential patterns mining methods, this paper proposed a standard permutation based method called SP-DSP. This method first mined frequent sequential patterns by the GSP algorithm, then the patterns whose Growth rate are less than the threshold were eliminated. Finally, the standard permutation method was used to compute the p-values of tested patterns. As a result, the number of false positive patterns can be controlled under the multiple hypothesis testing measures. The experiments showed that the SP-DSP algorithm can alleviate a lot of false positive patterns and retain as many true patterns as possible, which improves the accuracy of the downstream classification tasks.

Key words: discriminative sequential patterns mining; pattern assessment; multiple hypothesis testing; standard permutation testing

0 引言

序列数据指的是数据元素之间具备某种顺序关系的数据，例如网页浏览序列、蛋白质序列和人类语言都是常见的序列数据。在含有类型标签的序列数据中，某些序列模式在不同类别中出现的频率显著不同，这样的模式被称为差异序列模式[1]。差异序列模式在医学等许多应用中都有相当重要的价值[2~5]。

到目前为止，已经提出了一些有效的差异序列模式挖掘算法[6~8]。这些方法主要探讨了如何快速有效地挖掘差异序列模式，而没有关注挖掘到的模式的真假性，即这些算法返回的结果中会存在一定数量的假阳性差异序列模式。假阳性差异序列模式指的是在数据集中随机出现的并不能够反映总体特征的差异序列模式。采用假阳性差异序列模式做后续研究可能会得到错误的结果。可以采用统计显著性检验对挖掘结果进行质量评估，从而过滤假阳性差异序列模式。

在统计显著性检验中，一个结果的显著性是由它的 *p-value* 值度量的。*p-value* 值越小则说明该结果统计显著性越强。当仅有一个差异序列模式被检验时，如果它的 *p-value* 小于一个阈值α，那么称该差异序列模式在统计显著水平为α的条件下是统计显著的。在许多实际情况中，多个差异序列模式需要被同时检验，这样的检验称为多重假设检验。*FWER*(family wise error rate)[9]和 *FDR*(false discovery rate)[10]是多重假设检验中两个常用的控制假阳性结果数量的度量。

近年来，使用统计显著性检验方法评估数据挖掘结果已经得到了广泛研究。在非序列数据的差异模式挖掘任务中，Webb 提出了两种技术用以检验模式的真假性，分别是：留出方法和直接计算方法[11]。Liu 等人总结了关联规则挖掘中的几种多重假设检验算法，并将这些算法分成了三类[12]，其中基于置换检验的方法是最有效的。随后，一些改进的置换检验算法被相继提出并应用于差异模式挖掘[13, 14]。Komiyama 等人提出了两个方法 LAMP-EP 和 QT-LAMP-EP 分别控制结果的 *FWER* 度量和 *FDR* 度量[15]。以上方法在非序列数据的差异模式挖掘中都取得了非常好的效果。

最近，为了验证多重假设检验对差异序列模式挖掘任务的有效性，He 等人设计了一个基于直接计算的方法 DSPM-MTC 控制假阳性模式的数量[1]。DSPM-MTC 根据支持度服从超几何分布的特性直接计算得到每个差异序列模式相应的 *p-value*。鉴于置换检验方法相较于直接计算方法在非序列数据任务中更为有效[12]，本文提出了一个基于标准置换检验的差异序列模式挖掘算法，即 SP-DSP 算法。该算法首先运用 GSP 算法挖掘得到候选差异序列模式[16]，随后对原始数据进行标准置换检验并得到相应的置换检验零分布，最后由该零分布计算得到候选差异序列模式的 *p-value*，并运用 *FWER* 和 *FDR* 度量将结果中的假阳性差异序列模式数量控制在统计显著水平α下。本文的主要贡献如下：

a) 提出了一个基于标准置换检验的差异序列模式挖掘算法 SP-DSP，该算法能够将挖掘结果中假阳性差异序列模

收稿日期：2020-04-15；**修回日期：**2020-05-26 **基金项目：**贵州省教育厅青年科技人才成长项目(黔教合 KY 字［2017］250)；贵州省科技厅联合基金项目(黔科合 LH 字［2017］7069)

作者简介：吴军(1990-)，男，贵州遵义人，讲师，硕士，主要研究方向为数据挖掘、机器学习、生物信息学(wujun.myway@gmail.com)；欧阳艾嘉(1975-)，男，湖南娄底人，副教授，硕导，博士，主要研究方向为数据挖掘、并行计算；张琳(1984-)，女，贵州遵义人，副教授，硕士，主要研究方向为数据挖掘.

式数量控制在统计显著水平α下。

b) 通过真实数据集上的实验结果证明了 SP-DSP 算法能够过滤一定数量的假阳性模式，并且比 DSPM-MTC 算法能够保留更多的真差异序列模式。

c) 证明了运用多重假设检验能够提升差异序列模式挖掘算法结果的可信度。

1　基本定义

1.1　频繁序列模式

一条序列 s=<$a_1,a_2,\ldots,a_l$>是由字母表 I={$i_1,i_2,\ldots,i_{|I|}$}中的字母构成的一个有序线性表，其中 $a_j\in I$。对于序列 s_1=<$a_1,a_2,\ldots,a_l$>和序列 s_2=<$a'_1,a'_2,\ldots,a'_m$>，如果每一个 $a'_j\in s_2$ 也在序列 s_1 中且符合 s_1 的元素顺序，则称 s_2 是 s_1 的子序列，表示为 $s_2\subseteq s_1$。例如，给定一个序列 s=<i_1,i_3,i_5,i_6,i_8>，<i_1,i_3>和<i_5,i_8>均是 s 的子序列，<i_5,i_3>不是 s 的子序列，因为其不满足 s 的元素顺序。

给定一个序列数据集合 D={$t_1,t_2,\ldots,t_{|D|}$}，序列 s 在 D 中支持度定义为 D 中包含 s 的序列总数，即 $\sup(s,D)=|\{t|t\in D\wedge s\subseteq t\}|$。如果一条序列 s 的支持度超过了用户定义的阈值 *min_sup*，则该序列被称为频繁序列模式。

1.2　差异序列模式挖掘

对于含有类型标签的序列数据而言，一些序列在不同的类型标签中呈现显著频率差异，这样的序列被称为差异序列模式。其中，序列在不同类别中的差异性可以由许多差异性度量来衡量[17]。为了便于讨论，采用仅使用包含两种类型标签的序列数据，并分别用 D_+和 D_-表示两个序列数据集合。提出的方法可以轻易拓展到多个类型标签的序列数据。

非统计检验的差异序列模式挖掘算法通常可以分为两个步骤。首先运用频繁序列模式挖掘算法挖掘一定数量的候选差异序列模式；随后计算这些模式的差异性度量值，如果符合给定的阈值约束，则被认定为差异序列模式。

1.3　统计显著性检验

统计显著性检验包含两种假设：零假设和备择假设。差异序列模式挖掘任务的零假设是差异序列模式在 D_+和 D_-数据集中分布相同。在该任务中，每一个差异序列模式的统计显著性由 *p-value* 度量。*p-value* 的定义是:假设差异序列模式 s 在 D_+和 D_-里具有相同分布的前提下，获得一个与 s 同样极端或者更加极端的差异序列模式的概率。一个差异序列模式的 *p-value* 越小，则他在不同的类别里具有相同分布的可能性就越小。独立检验某个差异序列模式时，若它的 *p-value* 小于某个阈值α，则该差异序列模式被称作在统计显著水平α下是统计显著的差异序列模式。

差异序列模式挖掘算法通常会返回大量差异序列模式，运用独立检验方法会导致假阳性结果的增加，因此这种情景更适用于多重假设检验。在多重假设检验中，*FWER* 和 *FDR* 是两个常用的统计度量。其中，*FWER* 的定义是发现一个假阳性差异序列模式的概率；*FDR* 的定义是假阳性差异序列模式比例的期望值。*FWER* 可以用 Bonferroni 校正控制[9]，*FDR* 可以用 BH 方法控制[10]。

2　SP-DSP 算法

2.1　差异性度量

SP-DSP 算法采用 *Growth rate* 作为序列模式的差异性度量。给定一个频繁序列模式 s，其 *Growth rate* 计算公式为

$$\mathrm{grow}(s,D)=\frac{\sup(s,D_+)}{\sup(s,D_-)}\tag{1}$$

如果一个序列模式的 Grow(s,D)大于等于用户定义的差异阈值 β，则该模式被称为候选差异序列模式。

2.2　置换检验

SP-DSP 使用的标准置换检验含有五个步骤：

a) 根据具体的任务建立一个零假设，再选择一个在零假设和备择假设下具有不同值的统计度量，并挖掘 D_+数据集中的候选差异序列模式 R。SP-DSP 算法的零假设是差异序列模式在 D_+和 D_-中具有不同的分布，选用的统计度量为 *Growth rate*。

b) 随机交换 D_+和 D_-中的序列数据，得到置换序列数据集：D'_+和 D'_-。置换过程如图 1 所示，设 D 包含 8 条序列数据{$t_1,t_2,\ldots,t_8$}，其中前 5 条属于 D_+数据集，后 3 条属于 D_-数据集。随机生成一个置换序列：7,5,1,4,2,3,8,6，根据该序列，将 t_1 的标签分配给 t_7，t_2 的标签分配给 t_5，依此类推得到置换序列数据集合。

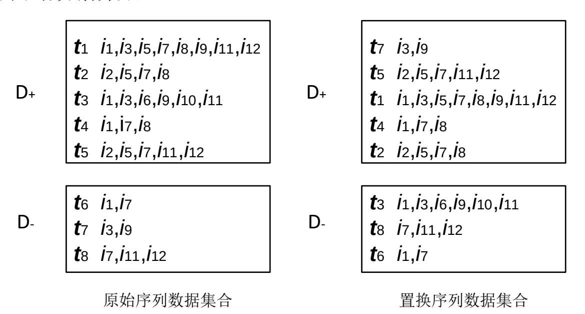


图 1　原始序列数据集合根据置换序列生成的置换序列数据集合

Fig. 1　The permuted sequential data set obtained from the original sequential data set with the permutation sequence

c) 挖掘 D'_+序列数据集中的差异序列模式，并将相应的统计度量值放入集合 G 中。

d) 重复第二步和第三步若干次后，用集合 G 中统计度量值构建该置换检验的零分布。通常执行的置换次数是 1000 次。

e) 将 D_+中的候选差异序列模式的统计度量值放置到上述零分布中计算得到置换检验 *p-value*，其计算公式为

$$\mathrm{p}(G,s)=\frac{|\{g_q\,|\,g_j\le g_q\wedge g_q\in G\}|}{|G|}\tag{2}$$

其中，g_j 指的是差异序列模式 s 在原始数据集合上的统计度量值。

2.3　多重假设检验

由置换检验计算得到候选差异序列模式的 *p-value* 后，SP-DSP 算法用 Bonferroni 校正和 BH 方法将挖掘结果 R 的 *FWER* 和 *FDR* 控制在统计显著水平 α 下。*FWER* 的计算公式如下：

$$\mathrm{FWER}(R,\alpha)=\{s\,|\,\mathrm{p}(G,s)\le\frac{\alpha}{|R|}\wedge s\in R\}\tag{3}$$

计算 *FDR* 时，需要先将 R 中差异序列模式的 *p-value* 按从小到大排序得到 R'={$s'_1,s'_2,\ldots,s'_{|R|}$}，随后可计算得到：

$$\mathrm{FDR}(R,\alpha)=\{s'_j\,|\,\mathrm{p}(G,s'_j)\le\frac{j\alpha}{|R'|}\wedge s'\in R'\}\tag{4}$$

2.4　SP-DSP 算法

SP-DSP 算法伪代码见算法 1，其详细的解释如下：

a) 在 D_+序列数据集中运用 gsp(D_+, *min_sup*)算法逐层挖掘 D_+数据集中支持度不小于 *min_sup* 的频繁序列模式，并将其放入到集合 *Freq* 中(第 a 行)；计算集合 *Freq* 中每一个频繁序列模式的 *Growth rate* 值，将超过阈值 β 的频繁序列模式放入集合 R 中，R 中的差异序列模式既是候选差异序列模式(第 b~e 行)。

b) 对于每一次置换 j，首先运用 permutate(D)方法进行类型标签的置换，得到置换数据集合 D'_+和 D'_-；随后，使用 gsp(D'_+, *min_sup*)算法挖掘 D'_+数据集中的频繁序列模式，并将其放入到集合 $Freq_per_j$ 中；接着，使用 com_sta($Freq_per_j$,

β)方法计算 *Freq_per*$_j$ 中每一个频繁序列模式的 *Growth rate* 值，并将超过 *β* 的 *Growth rate* 值放入到集合 G_j 中；最后将 G_j 中的 *Growth rate* 值并入集合 *G* 中(第 f~k 行)。最终，*G* 中所有的 *Growth rate* 值构成该置换检验的零分布。

c) 将集合 *R* 中每个候选差异序列模式的差异性度量值放置到零分布中计算出 *p-value* 值(第 l~n 行)；随后，根据每个模式的 *p-value* 值过滤得到非冗余的候选差异序列模式集合 *R*'(第 o 行)；最后，运用 Bonferroni 校正将 *R*'的 *FWER* 控制在统计显著水平 *α* 下，并将统计显著的差异序列模式保存到集合 R'_{FWER} 中；类似地，运用 BH 方法将 *R*'的 *FDR* 控制在统计显著水平 *α* 下，并将统计显著的差异序列模式保存到集合 R'_{FDR} 中(第 p~q 行)。

算法 1　SP-DSP(D, min_sup, α, β, num_per)

输入：序列数据集合 D={D+,D-}；最小支持度阈值 *min_sup*；统计显著水平 *α*；差异性度量阈值 *θ*；置换次数 *num_per*

输出：统计显著的差异序列模式集合 R'FWER 和 R'FDR

```
a) Freq ← gsp (D+, min_sup)
b) for each s in Freq
c) if Grow(s,D) > β
d)     R ← R∪{s}
e) end for
f) for j = 0 to num_per do
g)   D'+, D'- ← permutate(D)
h)   Freq_perj ← gsp(D'+ , min_sup)
i)   Gj ← com_sta(Freq_perj , θ)
j)   G ← G∪Gj
k) end for
l) for each s in R do
m)   s.p_value ← p(G, s)
n) end for
o) R' ← redundancy_filter(R)
p) R'FWER ← FWER(R', α)
q) R'FDR ← FDR(R', α)
```

算法 2 描述了去冗余方法 redundancy_filter(*R*)的详细步骤：对于每一个候选差异序列模式 *r*，先找到其相应的子序列模式集合 *Sub*；然后找到其中最小的 *p-value* 值 *min_p*，如果 *r* 的 *p-value* 值小于 *min_p*，则将其放入非冗余的候选差异序列模式集合 *R*'中；最后将 *R*'返回进行后续评估。

算法 2　redundancy_filter(*R*)

输入：带有 *p-value* 值的候选差异序列模式集合 *R*

输出：非冗余的候选差异序列模式集合 *R*'

```
a) for each r in R do
b) Sub ← getsubpatterns(r)
c) min_p ← min(Sub)
d)  if r.p_value < min_p
e)    R' ← R'∪{r}
f) end for
g) return R'
```

实验　为了检验 SP-DSP 算法的性能，本文在真实数据集上实施了大量对比实验。对比算法是 IMP 算法[6]、CGM 算法[8]和 DSPM-MTC 算法[1]。其中，DSPM-MTC 算法是基于多重假设检验的差异序列模式挖掘算法，IMP 算法和 CGM 算法是基于差异性度量的差异序列模式挖掘算法。同时，IMP 算法和 CGM 算法均使用了文献[1]中使用的去冗余方法。所有的相关实验均运行在一台配置为 2.40Ghz CPU 和 12GB 内存的电脑上。

2.5　实验数据

该实验选用了 4 个不同大小的序列数据集，分别是：Linux1_5[18]、Question[19]、WebKB[20]和 Reuters[21]。其中，Linux1_5、WebKB 和 Reuters 是多类别数据集，实验中只保留了这三个数据集中序列数量最多的两个类别。具体的数据集信息如表 1 所示。其中 l_{min}，l_{max} 和 l_{avg} 分别表示序列最短长度，序列最长长度和序列平均长度。

表 1　实验数据集

Tab. 1　The experimental data sets

数据集	|*I*|	|*D*|	l_{min}	l_{max}	l_{avg}
Linux1_5	426	1033	1	856	29.7
Question	3612	1731	4	29	10.2
WebKB	3151	2765	3	12	6.0
Reuters	15926	4436	4	815	87

2.6　实验结果

实验首先对比了 SP-DSP_{FDR}，DSPM-MTC_{FDR}，IMP 和 CGM 在相同的 *min_sup*，*α* 和 *β* 参数下不同数据集返回的差异序列模式的数量。实验结果如图 2 所示，从中可以明显看出：SP-DSP_{FDR} 和 DSPM-MTC_{FDR} 算法返回的结果数量小于 IMP 和 CGM 算法，其原因是差异性度量约束只关注了差异序列模式本身，而多重假设检验是对算法整个返回结果进行评估，所以基于多重假设检验的算法对结果质量的约束更为严格。

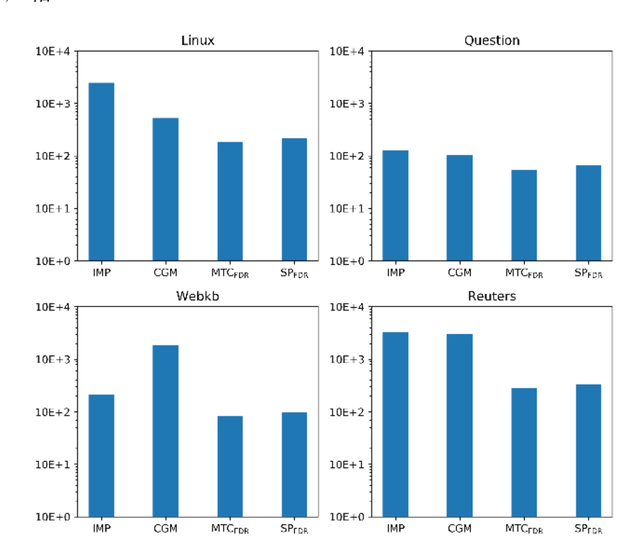


图 2　四种方法在各个数据集上返回的差异序列模式数量

Fig. 2　The number of discriminative sequential patterns returned from different algorithms on each data set

随后，实验对比了 SP-DSP 和 DSPM-MTC 算法在相同参数下使用 *FWER* 和 *FDR* 约束返回的差异序列模式的数量，实验结果如表 2 所示。从实验结果中可以看出在相同 *FDR* 或者 *FWER* 约束下，SP-DSP 算法返回的模式数量大于 DSPM-MTC 算法，这说明基于置换检验的方法比基于直接计算的方法能报告更多的结果数量。同时也能看出，同一种方法在 *FWER* 约束下报告的差异序列模式数量小于在 *FDR* 约束下报告的数量，这证明了 *FWER* 度量比 *FDR* 度量的约束更为严格。

表 2　SP-DSP 和 DSPM-MTC 算法在 FWER 或 FDR 约束下各个数据集返回的差异序列模式的数量

Tab. 2　The number of discriminative sequential patterns returned from the SP-DSP and DSPM-MTC methods under the *FWER* or *FDR* measure

	Linux1_5	Question	Webkb	Reuters
SP-DSP_{FDR}	214	67	96	336
SP-DSP_{FWER}	191	57	85	292
DSPM-MTC_{FDR}	186	54	84	278
DSPM-MTC_{FWER}	157	49	72	240

以上实验结果表明：相较于非多重假设检验方法 IMP 和 CGM，SP-DSP 能够过滤掉大量的模式；相较于多重假设检验方法 DSPM-MTC，SP-DSP 方法能保留更多的差异序列模式。

由于真实数据集中没有差异序列模式的 Groud truth 信息，无法根据上述结果直接说明 SP-DSP 方法相较于其他方法找到的差异序列模式准确性更高。为了证明挖掘算法的准确性，随后的实验将上述挖掘到的模式作为特征用于分类器进行分类预测任务[22]。分类任务之所以能够证明挖掘到的模式的准确性是因为真差异序列模式反映了不同类别数据集的分布差异性，从而对应了相应的类型标签。具体做法是，根据挖掘到的差异序列模式的数量，为数据集中每一条序列构造一个与该数量大小相同的向量作为特征表示，其中，如果某一条序列包含某一个模式，则该序列在该特征上的值为 1，反之，该序列在该特征上的值为 0。

考虑到不同分类方法的影响，实验使用了三种不同机制的分类方法：朴素贝叶斯(表示为 NB)，支持向量机(表示为 SVM)和全连接神经网络(表示为 MLP)。为了避免随机偶然性，每个分类方法都使用了五折交叉验证，并取十次预测结果的平均值作为最终的分类正确率。具体的实验结果如表 3~5 所示。

表 3 NB 分类方法在各个数据集上的正确率

Tab. 3 The classification accuracy on each data set returned from NB method

	Linux1_5	Question	WebKB	Reuters
IMP	0.692	0.835	0.572	0.802
CGM	0.754	0.842	0.616	0.704
DSPM-MTC$_{FDR}$	0.866	0.865	0.674	0.845
SP-DSP$_{FDR}$	**0.892**	**0.885**	**0.687**	**0.870**

表 4 SVM 分类方法在各个数据集上的正确率

Tab. 4 The classification accuracy on each data set returned from SVM method

	Linux1_5	Question	WebKB	Reuters
IMP	0.784	0.846	0.674	0.905
CGM	0.834	0.852	0.605	0.724
DSPM-MTC$_{FDR}$	0.862	0.872	0.722	0.926
SP-DSP$_{FDR}$	**0.896**	**0.887**	**0.738**	**0.942**

表 5 MLP 分类方法在各个数据集上的正确率

Tab. 5 The classification accuracy on each data set returned from MLP method

	Linux1_5	Question	WebKB	Reuters
IMP	0.792	0.862	0.685	0.902
CGM	0.838	0.876	0.624	0.746
DSPM-MTC$_{FDR}$	0.860	0.895	0.732	0.925
SP-DSP$_{FDR}$	**0.892**	**0.913**	**0.751**	**0.944**

从三种分类方法实验结果中可以得知：一方面，由 SP-DSP$_{FDR}$ 和 DSPM-MTC$_{FDR}$ 算法结果构成特征的分类正确率明显高于由 IMG 和 GCM 算法结果构成特征的分类正确率，这说明了多重假设检验的确过滤掉了许多假阳性差异序列模式。以 Question 数据集为例，IMG 和 GCM 算法挖掘结果存在<where, the> 模式，而 SP-DSP$_{FDR}$ 和 DSPM-MTC$_{FDR}$ 算法挖掘结果中只有<where>模式。Question 数据集中大量序列都存在定冠词 the，且定冠词 the 没有实义，因此用其作为特征很可能会导致错误分类。

另一方面，SP-DSP$_{FDR}$ 算法结果的正确率高于 DSPM-MTC$_{FDR}$ 算法结果的正确率，这体现了置换检验保留的更多的差异序列模式很可能是真差异序列模式。以 Question 数据集为例，SP-DSP$_{FDR}$ 算法结果中存在<what>和<what, in>模式，而 DSPM-MTC$_{FDR}$ 算法中只存在<what>模式，观察最终的错误分类结果发现，DSPM-MTC$_{FDR}$ 将正例类别中 6 条包含<what, in>模式的序列分到负例类别中，而 SP-DSP$_{FDR}$ 能够全部分对，这说明<what, in>模式应该是真差异序列模式。

此外，GCM 在 Webkb 数据集和 Reuters 数据集中低准确率现象说明假阳性模式对后续任务的严重误导性。同时，IMG 和 GCM 算法对不同的分类方法较为敏感，因为 IMG 和 GCM 算法会得到更多的干扰特征。

3 结束语

为了提升差异序列模式挖掘任务的准确率，提出了一个基于标准置换检验的算法 SP-DSP。真实数据集上的实验结果证明了运用多重假设检验能够提升差异序列模式挖掘算法结果的可信度。同时，相较于基于直接计算的多重假设检验方法 DSPM-MTC，SP-DSP 算法能够尽可能多的保留真差异序列模式。由于标准置换检验的随机性，SP-DSP 算法返回的统计显著的差异序列模式数量会有波动，本实验采用运行十次算法得到结果的平均值作为最终结果，后续工作将研究边界模式的舍取问题。

参考文献:

[1] He Zengyou, Zhang Simeng, Wu Jun. Significance-based discriminative sequential pattern mining [J]. Expert Systems with Applications, 2019, 122 (1): 54-64.

[2] Ghosh S, Li Jinyan, Cao Longbin, *et al.* Septic shock prediction for ICU patients via coupled HMM walking on sequential contrast patterns [J]. Journal of Biomedical Informatics, 2017, 66 (1): 19-31.

[3] 高增，郭均鹏. 考虑价格的跨种类模糊序列模式挖掘算法 [J]. 计算机应用研究, 2018, 35 (1): 39-42, 47. (Gao Zeng, Guo Junpeng. Cross-category fuzzy sequential pattern mining algorithm with consideration of price [J]. Application Research of Computers, 2018, 35 (1): 39-42, 47.)

[4] 张光兰，杨秋辉，程雪梅，等. 序列模式挖掘在通信网络告警预测中的应 [J]. 计算机科学, 2018, 45 (S2): 535-538, 563. (Zhang Guanglan, Yang Qiuhui, Cheng Xuemei, *et al.* Application of sequence pattern mining in communication network alarm prediction [J]. Computer Science, 2018, 45 (S2): 535-538, 563.)

[5] Liu Lu, Duan Lei, Yang Hao, *et al.* Mining distinguishing customer focus sets for online shopping decision support [C]// Proc of the 12th International Conference on Advanced Data Mining and Applications. Switzerland: Springer press, 2016: 50-64.

[6] Zheng Zhigang, Wei Wei, Liu Chunming, *et al.* An effective contrast sequential pattern mining approach to taxpayer behavior analysis [J]. World Wide Web, 2016, 19 (4): 633-651.

[7] He Zengyou, Gu Feiyang, Zhao Can, *et al.* Conditional discriminative pattern mining: concepts and algorithms [J]. Information Sciences, 2017, 375 (1): 1-15.

[8] Ji Xiaonan, Bailey J, Dong Guozhu. Mining minimal distinguishing subsequence patterns with gap constraints [J]. Knowledge and Information Systems, 2007, 11 (3): 259-286.

[9] Bland J, Altman D. Multiple significance tests: The bonferroni method [J]. British Medical Journal, 1995, 310 (6): 170-176.

[10] Benjamini Y, Krieger A, Yekutieli D. Adaptive linear step-up procedures that control the false discovery rate [J]. Biometrika, 2006, 93 (3): 491-507.

[11] Webb G. Discovering significant patterns [J]. Machine Learning, 2007, 68 (1): 1-33.

[12] Liu Guimei, Zhang Haojun, Wong L. Controlling false positives in association rule mining [J]. Proceedings of the VLDB Endowment, 2011, 5 (2): 145-156.

[13] Leonardo P, Fabio V. Efficient mining of the most significant patterns with permutation testing [C]// Proc of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. London: ACM press, 2018: 2070-2079.

[14] 吴军，段琼，张琳，等. 磷酸化基序精确置换检验 p-value 的计算方法 [J]. 中国科学：信息科学, 2017, 47 (10): 84-98. (Wu Jun, Duan Qiong, Zhang Lin, *et al.* Computing exact permutation p-values for phosphorylation motifs [J]. Scientia Sinica Informationis, 2017, 47 (10): 1334-1348.)

[15] Komiyama J, Ishihata M, Arimura H, *et al.* Statistical emerging pattern mining with multiple testing correction [C]// Proc of the 23th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Halifax: ACM press, 2017: 897-906.

[16] Srikant R, Agrawal R. Mining sequential patterns: generalizations and performance improvements [C]// Proc of the 6th International Conference on Extending Database Technology. Heidelberg: Springer press, 1996: 1-17.

[17] Liu Xiaoqing, Wu Jun, Gu Feiyang, *et al.* Discriminative pattern mining and its applications in bioinformatics [J]. Briefings in Bioinformatics, 2015, 16 (5): 884-900.

[18] Dua D, Graff C. UCI machine learning repository [EB/OL]. (2007) [2018-09-24]. http://archive. ics. uci. edu/ml.

[19] Kim Y. Convolutional Neural Networks for Sentence Classification [C]// Proc of the 18th Conference on Empirical Methods in Natural Language Processing. Doha: ACL press, 2015: 1745-1751.

[20] Craven M, Dipasquo D, Freitag D, *et al.* Learning to construct knowledge bases from the world wide web [J]. Artificial Intelligence, 2000, 118 (1): 69-113.

[21] Cardoso C. Improving methods for single-label text categorization [D]. Lisboa: Instituto Superior Técnico, 2007.

[22] Fradkin D, Mörchen F. Mining sequential patterns for classification [J]. Knowledge and Information Systems, 2015, 45 (3): 731-749.